WHAT IS CLAIMED IS:

1. A data processing system, comprising:

a memory mapped non-volatile memory region in including one or more executable instructions to initialize the data processing system;

a non-memory mapped non-volatile memory region, the non-memory mapped non-volatile memory region including one or more additional executable instructions to initialize the data processing system; and

a memory region in communication with at least one of the memory mapped non-volatile memory region and the non-memory mapped static memory region.

2. The system of claim 1, wherein the memory mapped static memory region comprises a first integrated circuit including a hardware mechanism to enable access to the one or more executable instructions using a processor memory read.

3. The system of claim 2, wherein the non-memory mapped non-volatile memory region comprises a second integrated circuit separate from the first integrated circuit.

4. The system of claim 3, wherein the second integrated circuit does not include a hardware mechanism to enable access

to the one or more additional executable instructions using a processor memory read.

5. The system of claim 1, wherein at least one of the memory mapped non-volatile memory region and the non memory-mapped non-volatile memory region is implemented as flash memory.

6. The system of claim 1, wherein the memory region comprises cache memory.

7. The system of claim 6, further including a microprocessor, and wherein the cache memory is integrated with the microprocessor.

8. The system of claim 6, wherein the cache memory is partitioned to include a first region to store data and a second region to store one or more pages.

9. The system of claim 8, wherein the cache memory is partitioned to further include a third region to store one or more page tables.

10. A method of initializing a computer system, comprising:

(a) accessing memory mapped firmware using a processor memory read;

(b) executing an instruction included in the memory mapped firmware;

(c) copying a page including another instruction from non-memory mapped firmware to another memory region; and

(d) executing the another instruction.

11. The method of claim 10, wherein actions (a) through (d) are performed prior to initializing a main memory of the computer system.

12. The method of claim 10, further comprising prior to (c), initializing at least a portion of a cache memory as random access memory.

13. The method of claim 10, further comprising prior to (c):

(b)(1) attempting to access a desired page of data stored in the another memory;

(b)(2) receiving a page fault signal indicating that the desired page of data is not stored in the another memory; and

(b)(3) copying the desired page of data from non-memory mapped firmware to the another memory.

14. The method of claim 13, further comprising receiving a fault serviced signal indicating that the desired of page of data is in the another memory.

15. The method of claim 14, further comprising executing an instruction included in the desired page of data stored in the another memory.

16. A chipset, comprising:

a memory mapped non-volatile memory region in including one or more executable instructions to initialize the data processing system;

a non-memory mapped non-volatile memory region, the non-memory mapped non-volatile memory region including one or more additional executable instructions to initialize the data processing system; and

a memory region in communication with at least one of the memory mapped non-volatile memory region and the non-memory mapped static memory region.

17. The chipset of claim 16, wherein the memory mapped static memory region comprises a first integrated circuit included in the chipset, the first integrated circuit including a hardware mechanism to enable access to the one or more executable instructions using a processor memory read.

18. The chipset of claim 17, wherein the non-memory mapped non-volatile memory region comprises a second integrated circuit included in the chipset, the second integrated circuit separate from the first integrated circuit.

19. The chipset of claim 18, wherein the second integrated circuit does not include a hardware mechanism to enable access to the one or more additional executable instructions using a processor memory read.

20. The chipset of claim 18, wherein at least one of the memory mapped non-volatile memory region and the non memory-mapped non-volatile memory region is implemented as flash memory.

21. The chipset of claim 16, wherein the chipset is in communication with a microprocessor.

22. The chipset of claim 16, wherein the memory region comprises cache memory.

23. The chipset of claim 22, wherein the cache memory is configurable to be partitioned to include a first region to store data and a second region to store one or more pages.

24. The chipset of claim 23, wherein the cache memory is configurable to be partitioned to further include a third region to store one or more page tables.

25. An article of manufacture comprising a machine accessible medium containing code having instructions that, when executed, cause the machine to:

(a) access memory mapped firmware using a processor memory read;

(b) execute an instruction included in the memory mapped firmware;

(c) copy a page including another instruction from non-memory mapped firmware to another memory region; and

(d) execute the another instruction.

26. The article of claim 25, wherein the instructions cause the machine to perform (a) through (d) prior to initializing a main memory of the computer system.

27. The article of claim 25, the instructions further causing the machine to initialize at least a portion of a cache memory as random access memory prior to (c).

28. The article of claim 25, the instructions further causing the machine to, prior to (c):

(b)(1) attempt to access a desired page of data stored in the another memory; and

(b)(2) upon receiving a page fault signal indicating that the desired page of data is not stored in the another memory, copy the desired page of data from non-memory mapped firmware to the another memory.

29. The article of claim 28, the instructions further causing the machine to receive a fault serviced signal indicating that the desired of page of data is in the another memory.

30. The article of claim 29, the instructions further causing the machine to execute an instruction included in the desired page of data stored in the another memory.